PHIL 7001: Fundamentals of AI, Data, and Algorithms

Week 9
Reinforcement Learning

Maomei Wang
Email: mmmw@connect.hku.hk

Outline

Contents

Reinforcement Learning

What is RL

RL is a kind of ML that has the following two main differences from classical ML:

- **Goal.** It aims to learn how to make optimal sequential decisions. In other words, it is a process that machines *learn to decide* in a dynamic process.
- **Data.** The data received are neither labeled nor non-labeled but are the rewards of the decisions.

Reinforcement learning is learning what to do — how to map situations to decisions — so as to maximize a numerical reward.

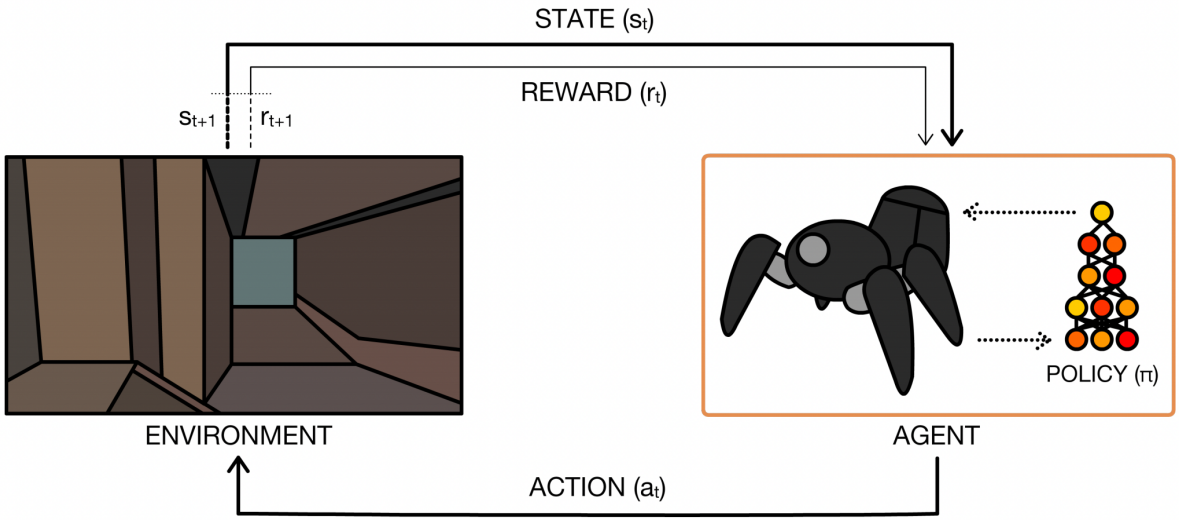
Examples of RL

- Chess player: what's the next move? The choice is informed both by planning — anticipating possible replies and counterreplies — and by immediate, intuitive judgments of the desirability of particular positions and moves.
- Mobile robot: what's the next action? E.g., when to enter a new room in search of more trash to collect or start trying to find its way back to its battery recharging station.

Can we use supervised learning to learn the above?

Main elements of RL I

An RL agent observes a state s_t and takes an action a_t, which leads a state transition from s_t to s_{t+1}.
The environment provides a reward r_{t+1} to the agent as feedback.
The goal of RL is to learn a policy π that maximizes the expected return (cumulative, discounted reward).



Main elements of RL II

- Policy. A function
$$\pi : \mathcal{S} \to \Delta(\mathcal{A})$$
such that $\pi(s)$ is a probability distribution over actions.
- Reward function R (and discount factor γ).
- Trajectory τ is a sequence of states, actions and rewards, i.e.,
$$\tau = (s_0, a_0, r_0, s_1, a_1, r_1, \dots).$$
- Value function. $Q^\pi(s, a)$, the value of taking action a at sate s under policy π.
- (Optional) Model of the environment. How does state s_t transfer to s_{t+1} after taking a_t?

What is DRL

Deep Learning + Reinforcement Learning = DRL

Applying DL techniques within RL, we get Deep Reinforcement Learning (DRL).
In particular, DNNs are employed to do function approximation. (What functions to be approximated?) And learning is done by varients of SGD.

Why Deep

Classical RL were limited to fairly low-dimensional problems. The most important property of deep learning is that DNNs can automatically find representations (features) of high-dimensional data (e.g., images, text, and audio). Thus it is natural to employ DL techniques to do RL.

- E.g., Mnih *et al.* (2015) Human-Level Control through Deep Reinforcement Learning. *Nature, 518*(7540):529–533.
- Do not need the handcrafted rules anymore. Learn to play a range of Atari 2600 video games at a superhuman level, from nothing but the video input.

Examples of DRL I

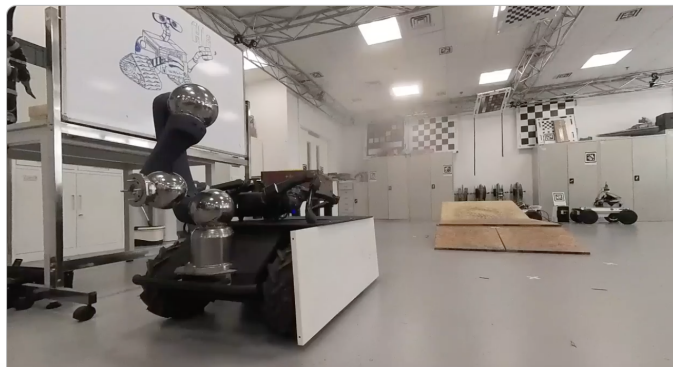
Drone-racing champions outpaced by AI

An autonomous drone has competed against human drone-racing champions – and won. The victory can be attributed to savvy engineering and a type of artificial intelligence that learns mostly through trial and error.

https://www.youtube.com/watch?v=fBiataDpGIo

Examples of DRL II



https://robot-parkour.github.io/

Examples of DRL III

Ke Jie, Humanity's Last Hope, Loses to AlphaGo by Half a Point
Link: `https://www.sixthtone.com/news/1000242`.

Types of DRL: Model-free vs. Model-based

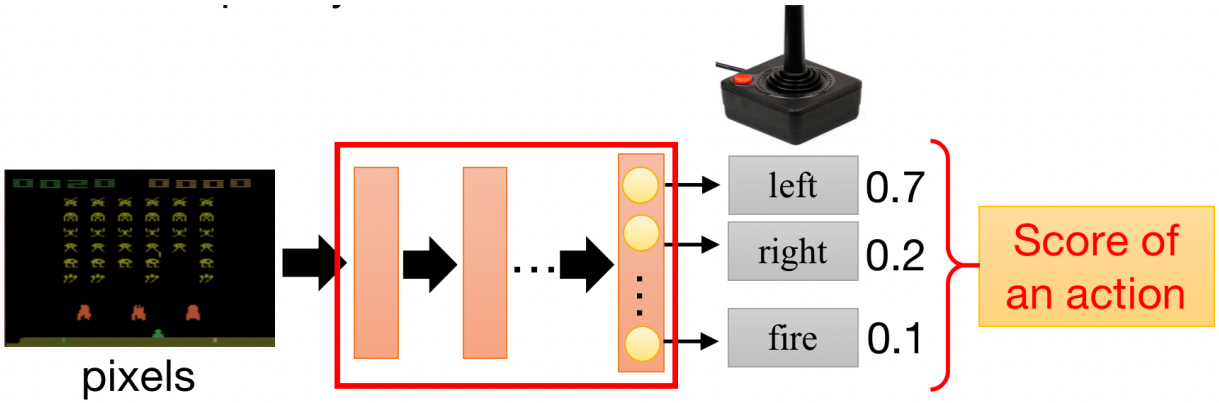
Model: Markov decision processes (MDP) — based on some assumptions about how state transition goes.

- Model-free:
  - Policy-based DRL: approximate the optimal policy function.
  - Value-based DRL: approximate the optimal value function $Q^\star(s, a)$ e.g., DQN.
- Model-based: based on the MDP model of the environment and plan using the model.

Using DNNs to approximate policy, value function, or (and) model.

DQN and Atari 2600

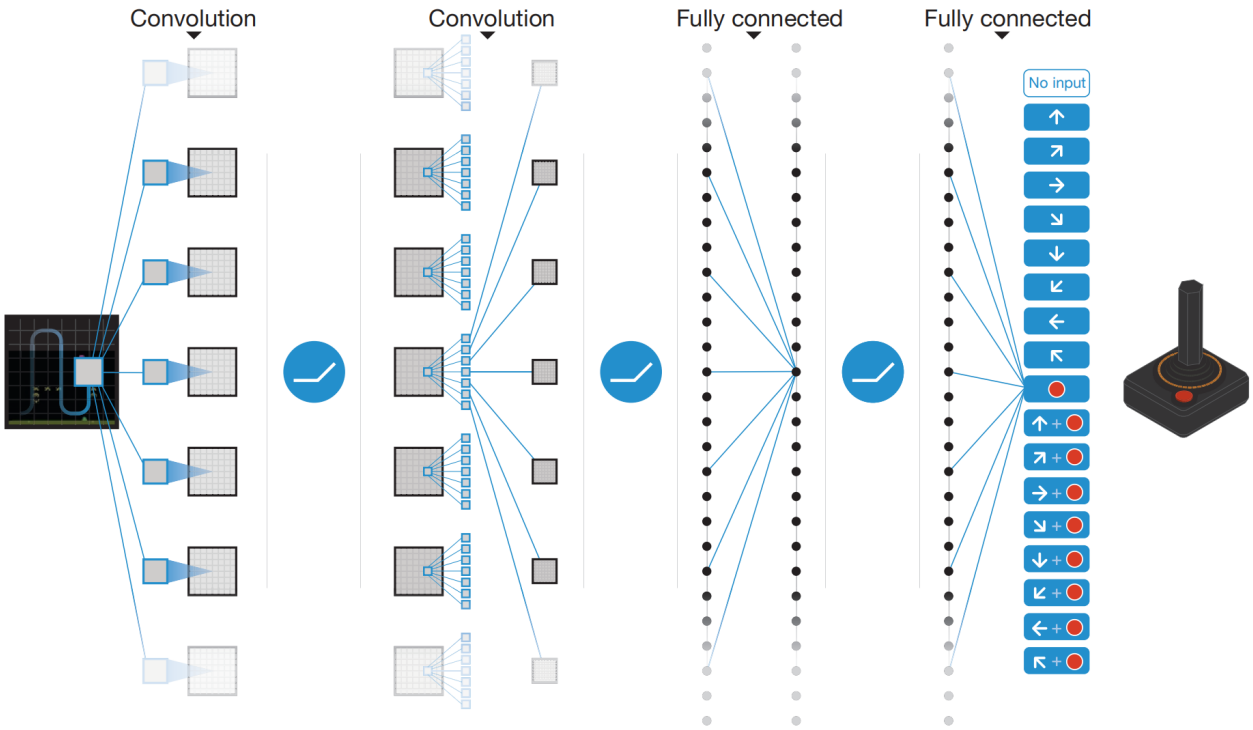
Approximate $Q^\star(s, a)$, and then use this function to make decision.



Paper: Mnih *et al.* (2015) Human-Level Control through Deep Reinforcement Learning. *Nature, 518*(7540):529–533.

CNN: a kind of neural network that can "watch"

- Convolutional Neural Network (CNN) includes layers called convolutional layer that are useful for processing and analyzing visual data such as images or videos. (They may also consist of pooling layers though.)
- An example CNN:



Source: Mnih *et al.* (2015).

- The exact architecture, including the number and size of convolutional layers, depends on the specific problem at hand.

MDP

- Markov Chain assumption:

$$p(s_{t+1}|s_t) = p(s_{t+1}|s_0, s_1, \ldots, s_t)$$

- MDP is a tuple $< \mathcal{S}, \mathcal{A}, P, R, \gamma >$ where
  - $\mathcal{S}$ denotes a set of states
  - $\mathcal{A}$ denotes a set of actions
  - $(\star)$ P denotes a probability function such that

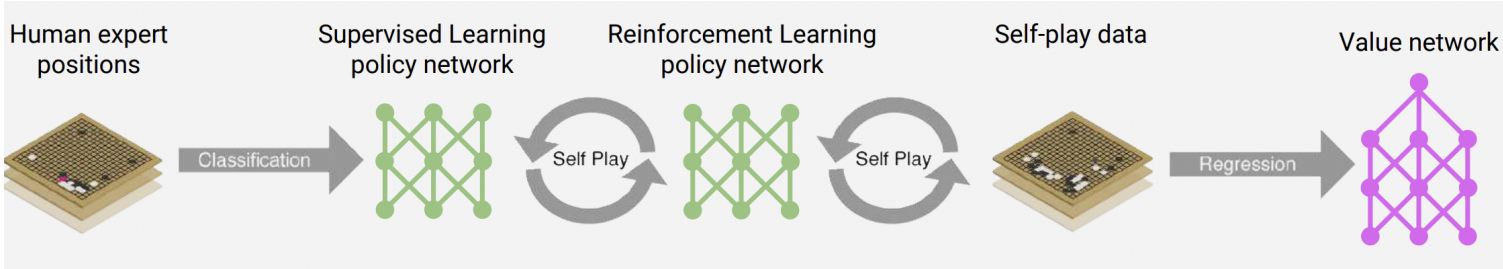
$$P(s'|s, a) = P(s_{t+1} = s'|s_t = s, A_t = a)$$

  - R denotes the reward function
  - γ represents the reward discount factor, such that $\gamma \in [0, 1]$ is a weight on reward.

The MDP model generalizes the Markov Chain assumption and assumes that the state at s_{t+1} only depends on the pair (s_t, a_t).

> Is this assumption reasonable?

AlphaGo



Source: Google DeepMind

Paper: Silver *et al.* (2016). Mastering the game of Go with deep neural networks and tree search. *nature, 529*(7587), 484-489.

- Supervised learning of policy networks: 12-layer CNN
- Reinforcement learning of policy networks: 12-layer CNN
- Reinforcement learning of value networks: 12-layer CNN.

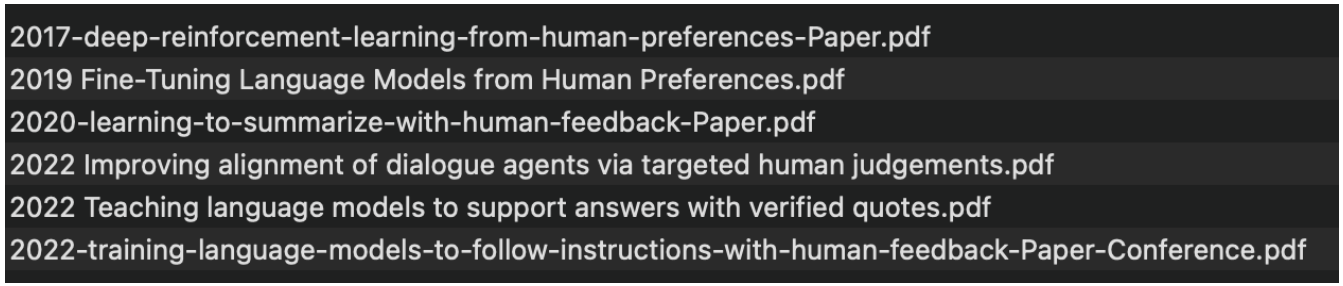
Applications

- Game Playing
- Robotics
- Autonomous Vehicles: e.g., self-driving cars.
- Recommendation Systems: personalize recommendations in various domains, such as e-commerce, content streaming platforms, and online advertising.
- Healthcare: DRL can assist in medical diagnosis, treatment planning, and drug discovery.
- Finance: make intelligent decisions for buying, selling, and managing financial assets.
- etc.

Can you give examples of real-life DRL products?

Alignment and LLMs

RLHF: Reinforcement Learning from Human Feedback



Large language models (LLMs) can generate outputs that are untruthful, toxic, or simply not helpful to the user. In other words, these models are not aligned with their users.
By RLHF, LLMs can learn from human expertise and preferences, improving its ability to generate more accurate, relevant, and contextually appropriate responses.

It inherits problems with DL

The development of RL benefits from the employment of DL techniques, but also inherits drawbacks of them.

- Explainability. Hard to understand because of the black box nature of DNNs.
- Data. Difficult to train, require a large amount of data
- Overfit. May overfit and may not generalize well to new situations.
- etc.

Other problems: reward specification

Unlike other ML techniques, which use loss functions to evaluate the behaviors of machines, in RL we use reward functions to do so. But where can we get the reward function? Consider:

- Subjectivity: Different people may have different opinions on what the agent should optimize for.
- Incomplete or Incorrect Rewards: Designing a reward function that captures all relevant aspects of the problem can be complex.
- Sparsity. In some scenarios, providing meaningful rewards can be challenging due to sparse feedback.
- Reward Trade-offs: Sometimes, there are trade-offs between different objectives, and it becomes challenging to design a reward function that balances these trade-offs appropriately. E.g., self-driving: quick, safe, and ...?
- Changing Objectives: The desired behavior or objectives may change over time. E.g., what do consumers want?
- etc.

Exploration vs. Exploitation

To obtain a lot of reward, a reinforcement learning agent must prefer actions that it has tried in the past and found to be effective in producing reward. But to discover such actions, it has to try actions that it has not selected before. The agent has to exploit what it has already experienced in order to obtain reward, but it also has to explore in order to make better action selections in the future. The dilemma is that neither exploration nor exploitation can be pursued exclusively without failing at the task.

- The exploration-exploitation dilemma has been intensively studied by mathematicians for many decades, yet remains unresolved.

(Sutton, R. S., & Barto, A. G. 2018)

More problems

Since RL is supposed to be applied to decision-making, DRL leads to even more ethical worries...
A perhaps extreme example:

https://www.youtube.com/watch?v=W0_DPi0PmF0

Describe possible or real problems arising from applying DRL in the real world.

Main points: RL

1. RL is a process in which machines learn how to make sequential decisions.
2. During RL, the data are rewards and the goal is to maximize the expected overall reward.
3. We can use DL techniques to make RL deep, thus DRL, in the sense that we can input raw data to learn.
4. DRL is a process of approximating a value function, a policy function, or (and) a model.
5. DRL has many real-life applications, e.g., it is employed in LLMs to do alignment.
6. DRL inherits general problems with DL, but also has other problems, such as exploration vs. exploitation dilemma, and reward specification problem. In particular, since it involves decision-making, it may lead to more ethical worries.

References

- Arulkumaran, K., Deisenroth, M. P., Brundage, M., & Bharath, A. A. (2017). Deep reinforcement learning: A brief survey. *IEEE Signal Processing Magazine, 34*(6), 26-38.
- Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., ... & Hassabis, D. (2015). Human-level control through deep reinforcement learning. *nature, 518*(7540), 529-533.
- Silver, D., (2016) David Silver ICML 2016. Retrieved from `https://www.davidsilver.uk/wp-content/uploads/2020/03/deep_rl_tutorial_small_compressed.pdf`.
- Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., ... & Hassabis, D. (2016). Mastering the game of Go with deep neural networks and tree search. *nature, 529*(7587), 484-489.
- Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An introduction.* MIT press.